64000

HP64000 Logic Development System

Model 64855A Assembler Supplement 8051

CERTIFICATION

Hewlett-Packard Company certifies that this product met its published specifications at the time of shipment from the factory. Hewlett-Packard further certifies that its calibration measurements are traceable to the United States National Bureau of Standards, to the extent allowed by the Bureau's calibration facility, and to the calibration facilities of other International Standards Organization members.

WARRANTY

This Hewlett-Packard system product is warranted against defects in materials and workmanship for a period of 90 days from date of installation. During the warranty period, HP will, at its options, either repair or replace products which prove to be defective.

Warranty service of this product will be performed at Buyer's facility at no charge within HP service travel areas. Outside HP service travel areas, warranty service will be performed at Buyer's facility only upon HP's prior agreement and Buyer shall pay HP's round trip travel expenses. In all other cases, products must be returned to a service facility designated by HP

For products returned to HP for warranty service. Buyer shall prepay shipping charges to HP and HP shall pay shipping charges to return the product to Buyer. However, Buyer shall pay all shipping charges, duties, and taxes for products returned to HP from another country.

HP warrants that its software and firmware designated by HP for use with an instrument will execute its programming instructions when properly installed on that instrument. HP does not warrant that the operation of the instrument, or software, or firmware will be uninterrupted or error free.

LIMITATION OF WARRANTY

The foregoing warranty shall not apply to defects resulting from improper or inadequate maintenance by Buyer, Buyer-supplied software or interfacing, unauthorized modification or misuse, operation outside of the environmental specifications for the product, or improper site preparation or maintenance.

NO OTHER WARRANTY IS EXPRESSED OR IMPLIED. HP SPECIFICALLY DISCLAIMS THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

EXCLUSIVE REMEDIES

THE REMEDIES PROVIDED HEREIN ARE BUYER'S SOLE AND EXCLUSIVE REMEDIES. HP SHALL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER BASED ON CONTRACT, TORT, OR ANY OTHER LEGAL THEORY.

ASSISTANCE

Product maintenance agreements and other customer assistance agreements are available for Hewlett-Packard products.

For any assistance, contact your nearest Hewlett-Packard Sales and Service Office.

CW&A 9/79

FOLD HERE

HEWLETT PACKARD

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

BUSINESS REPLY CARD
FIRST CLASS PERMIT NO. 1303 COLORADO SPRINGS, COLORADO

POSTAGE WILL BE PAID BY ADDRESSEE

HEWLETT-PACKARD
Logic Product Support Dept.
Attn: Technical Publications Manager
Centennial Annex - D2
P.O. Box 617
Colorado Springs, Colorado 80901-0617

FOLD HERE

Your cooperation in completing and returning this form will be greatly appreciated. Thank you.

READER COMMENT SHEET

Operating Manual, Model 64855A
Assembler Supplement 8051
64855-90902, December 1980

Your comments are important to us. Please answer this questionaire and return it to us. Circle the number that best describes your answer in questions 1 through 7. Thank you.

1. The information in this book is complete:

Doesn't cover enough (what more do you need?)	1	2	3	4	5	Covers everything

2. The information in this book is accurate:

Too many errors	1	2	3	4	5	Exactly right

3. The information in this book is easy to find:

I can't find things I need	1	2	3	4	5	I can find info quickly

4. The Index and Table of Contents are useful:

Helpful	1	2	3	4	5	Missing or inadequate

5. What about the "how-to" procedures and examples:

No help	1	2	3	4	5	Very helpful
Too many now	1	2	3	4	5	I'd like more

6. What about the writing style:

Confusing	1	2	3	4	5	Clear

7. What about organization of the book:

Poor order	1	2	3	4	5	Good order

8. What about the size of the book:

too big/small	1	2	3	4	5	Right size

Comments: _____

_____

_____

_____

Particular pages with errors?

_____

Name (optional): _____
Job title: _____
Company: _____
Address: _____

Note: If mailed outside U.S.A., place card in envelope. Use address shown on other side of this card.

FOLD HERE

HEWLETT PACKARD

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

BUSINESS REPLY CARD
FIRST CLASS PERMIT NO. 1303 COLORADO SPRINGS, COLORADO

POSTAGE WILL BE PAID BY ADDRESSEE

HEWLETT-PACKARD
Logic Product Support Dept.
Attn: Technical Publications Manager
Centennial Annex - D2
P.O. Box 617
Colorado Springs, Colorado 80901-0617

FOLD HERE

Your cooperation in completing and returning this form will be greatly appreciated. Thank you.

READER COMMENT SHEET

Operating Manual, Model 64855A
Assembler Supplement 8051
64855-90902, December 1980

Your comments are important to us. Please answer this questionaire and return it to us. Circle the number that best describes your answer in questions 1 through 7. Thank you.

1. The information in this book is complete:

Doesn't cover enough (what more do you need?)	1	2	3	4	5	Covers everything

2. The information in this book is accurate:

Too many errors	1	2	3	4	5	Exactly right

3. The information in this book is easy to find:

I can't find things I need	1	2	3	4	5	I can find info quickly

4. The Index and Table of Contents are useful:

Helpful	1	2	3	4	5	Missing or inadequate

5. What about the "how-to" procedures and examples:

No help	1	2	3	4	5	Very helpful
Too many now	1	2	3	4	5	I'd like more

6. What about the writing style:

Confusing	1	2	3	4	5	Clear

7. What about organization of the book:

Poor order	1	2	3	4	5	Good order

8. What about the size of the book:

too big/small	1	2	3	4	5	Right size

Comments: _____

_____

_____

_____

Particular pages with errors?

_____

Name (optional): _____
Job title: _____
Company: _____
Address: _____

Note: If mailed outside U.S.A., place card in envelope. Use address shown on other side of this card.

Printing History

Each new edition of this manual incorporates all material updated since the previous edition. Each new or revised page is indicated by a revision (rev) date. Manual change sheets are issued between editions, allowing you to correct or insert information in the current edition.

The part number on the back cover changes only when each new edition is published. Minor corrections or additions may be made as the manual is reprinted between editions.

First Printing ..... December 1980 (Part Number 64855-90902)

Assembler Supplement
8051

© COPYRIGHT HEWLETT-PACKARD COMPANY 1980
LOGIC SYSTEMS DIVISION
COLORADO SPRINGS, COLORADO, U.S.A.

ALL RIGHTS RESERVED

The information in this supplement has been checked for accuracy and is believed to be correct; however, no responsibility is assumed for inaccuracies. When discrepancies are noted, refer to the manufacturer's Microprocessor Programming Manual for clarification.

Table of Contents

Chapter 1: General Information

Introduction 1-1
Architecture 1-1
- Memory Segmentation 1-1
- Program Memory 1-2
- Data Memory 1-2
- Bit Addressing 1-2
- General Purpose Registers 1-2
- Stack Register 1-2
- Hardware Registers 1-3
- Program Status Word (PSW) 1-4
- Timer and Counter 1-5
- I/O Ports 1-6
- Serial I/O Port 1-7
- Interrupt Control 1-7
- Reset 1-9

Addressing Modes 1-10
- Indirect Addressing 1-10
- Immediate Data 1-10
- Data Addressing 1-10
- Code Addressing 1-10

Chapter 2: Rules and Conventions

Introduction 2-1
Operands 2-1
- Special Assembler Symbols 2-1
- Indirect Addressing 2-2
- Immediate Data 2-3
- Data Addressing 2-3
- Bit Addressing 2-4
- Code Addressing 2-4
- Symbols 2-6
- Special Operators 2-6

Predefined Symbols 2-7
- Reserved Words 2-7
- Prefined Data (Byte) Addresses 2-7
- Predefined Bit Addresses 2-7
- Predefined Code Addresses 2-7

Table of Contents (Cont'd)

Chapter 3: Special Pseudo Instructions

Introduction 3-1
BIT Instruction 3-2
BSEG Instruction 3-3
CSEG Instruction 3-4
DATA Instruction 3-5
DSEG Instruction 3-6
DBIT Instruction 3-7
DB Instruction 3-8
DW Instruction 3-9
DS Instruction 3-10
SET Instruction 3-11
XDATA Instruction 3-12
XSEG Instruction 3-13

List of Illustrations

1-1. Bits in Program Status Word 1-4
1-2. Bits in Timer Control Register 1-5
1-3. Bits in Port 3 Register 1-6
1-4. Bits in SCON Register 1-7
1-5. Descriptions of Interrupt Enable and Interrupt Priority Bits 1-8

List of Tables

1-1. Selection of General Purpose Register Banks 1-2
1-2. Hardware Registers 1-3
1-3. I/O Port Addressing 1-6
1-4. Labels and Locations of Interrupting Devices 1-8
1-5. Register Values At Reset 1-9

2-1. Special Assembler Symbols 2-2

Chapter 1

General Information (8051)

Introduction

This chapter briefly explains the 8051 microcomputer memory segmentation, registers, and addressing modes. For detailed descriptions and the instruction set with the object code it generates, refer to the manufacturer's users guide.

NOTE

The assembler directive for the 8051 is "8051".

Architecture

Memory Segmentation

The 8051 has four separate address segments: code address, internal data address, external data address, and bit address. The first three have physically separate memories. Bit address shares locations in the data address space. For the programmer, nothing distinguishes on-chip code from off-chip code. A 16-bit program counter addresses both memories with the same instruction fetch time.

Program Memory

The 8051 has no resident program memory; therefore up to 64k of off-chip code address memory can be added by the user.

Data Memory

The 8051 has 128 bytes of on-chip RAM for internal data address and bit address space. Up to 64k bytes of external data can be addressed in off-chip RAM.

Bit Addressing

Bits in many of the hardware control registers are directly addressable with a bit selector (.). For example, to address bit 2 in the accumulator, use ACC.2. More information is included in Chapter 2, Bit Addressing.

General Purpose Registers

The 8051 has 32 general purpose registers, arranged in four banks of eight. The registers are located in locations 00H to 1FH of on-chip RAM (first 32 bytes). The bank that is currently active is accessed with symbols: R0, R1, R2, R3, R4, R5, R6, and R7. The banks are selected with register bank select bits RS0 and RS1 in the program status word. Table 1-1 shows how the banks are selected and their memory locations.

Table 1-1. Selection of General Purpose Register Banks

RS0	RS1	Bank	Memory Locations
0	0	0	00H-07H
0	1	1	08H-0FH
1	0	2	10H-17H
1	1	3	18H-1FH

Stack Register

The stack is an 8-bit register in on-chip RAM with LIFO storage. It holds the contents of the program counter during interrupts and subroutine calls. The programmer can also store data in and return data from the stack with the POP and PUSH instructions. The stack pointer contains the address on the top of the stack. It can contain any address in on-chip RAM memory, but it must not exceed 127. On reset, the stack pointer is set to 07H; therefore, the first byte pushed on the stack after reset is located at 08H.

Hardware Registers

The hardware registers are addressed with a numeric data address or a predefined symbol. Table 1-2 lists the predefined symbol, hexadecimal data address, and register name.

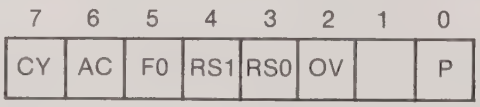
Table 1-2. Hardware Registers

Predefined Symbol	Data Address	Register Name
ACC	0E0H	ACCUMULATOR (Data address A)
B	0F0H	MULTIPLICATION
DPH	83H	DATA POINTER (High byte)
DPL	82H	DATA POINTER (Low byte)
IE	0A8H	INTERRUPT ENABLE
IP	0B8H	INTERRUPT PRIORITY
P0	80H	PORT 0
P1	90H	PORT 1
P2	0A0H	PORT 2
P3	0B0H	PORT 3
PSW	0D0H	PROGRAM STATUS WORD
SBUF	99H	SERIAL PORT BUFFER
SCON	98H	SERIAL PORT CONTROLLER
SP	81H	STACK POINTER
TCON	88H	TIMER CONTROL
TH0	8CH	TIMER 0 (High byte)
TH1	8DH	TIMER 1 (High byte)
TL0	8AH	TIMER 0 (Low byte)
TL1	8BH	TIMER 1 (Low byte)
TMOD	89H	TIMER MODE

Program Status Word (PSW)

The PSW contains seven status bits that provide status information on the 8051. Figure 1-1 explains each of the status bits. The bit selector to address each of the bits is shown in parenthesis.

BITS



CY: Carry flag indicates carry from bit 7 of ALU operands (PSW.7).
AC: Auxiliary carry flag indicates carry from bit 3 of addition operands (PSW.6).
F0: General purpose status flag (PSW.5).
RS1: Register bank select bit 1 (PSW.4).
RS0: Register bank select bit 0 (PSW.3).
OV: Overflow flag set by arithmetic operations (PSW.2).
BIT 1: User definable flag (PSW.1).
P: Accumulator parity. Hardware sets to 1 for odd number of 1's; otherwise reset to 0 (PSW.0).

Figure 1-1. Bits in Program Status Word

Timer and Counter

There are two independently programmable timers in the 8051. They have a 16-bit counter and are controlled by the timer mode register (TMOD) and timer control register (TCON). Figure 1-2 explains each of the bits in TCON and identifies the predefined bit address symbols. The bit selector to address each of the bits is shown in parenthesis.

BITS

7	6	5	4	3	2	1	0
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

TF1: Timer 1 overflow flag (TCON.7).
TR1: Timer 1 run control bit (TCON.6).
TF0: Timer 0 overflow flag (TCON.5).
TR0: Timer 0 run control bit (TCON.4).
IE1: Interrupt 1 edge flag (TCON.3).
IT1: Interrupt 1 type control bit (TCON.2).
IE0: Interrupt 0 edge flag (TCON.1).
IT0: Interrupt 0 type control bit (TCON.0).

Figure 1-2. Bits in Timer Control Register

I/O Ports

There are four 8-bit I/O ports in the 8051. The bits in each port relate to a specific pin on the 8051. All ports are buffered through latches and each port is addressable through a data address (byte) or 8-bit addresses (set of bits). Table 1-3 provides the hexadecimal data address and predefined symbol for each port. Ports 0 and 2 are used for external data and program addressing. In addition, port 0 receives input data from off-chip addressing. With no off-chip memory, ports 0 and 2 are bidirectional I/O ports. Port 1 is a general purpose, bidirectional I/O port.

Table 1-3. I/O Port Addressing

I/O Port	Predefined Symbol	Data Address
0	P0	80H
1	P1	90H
2	P2	0A0H
3	P3	0B0H

The external interrupt pins, external timer, external data memory read and write enables, and serial I/O port transmit and receive pins are part of port 3. Figure 1-3 explains each of the bits in port 3. The bit selector to address each of the bits is shown in parenthesis. Port 3 can be used as a bidirectional I/O port if there are no external interrupts, external data addressing, or serial I/O features.

BITS

7	6	5	4	3	2	1	0
RD	WR	T1	T0	INT1	INT0	TXD	RXD

RD: Read data for external memory (P3.7).
WR: Write data for external memory (P3.6).
T1: Timer/counter 1 external flag (P3.5).
T0: Timer/counter 0 external flag (P3.4).
INT1: Interrupt 1 input pin (P3.3).
INT0: Interrupt 0 input pin (P3.2).
TXD: Serial port transmit pin (P3.1).
RXD: Serial port receive pin (P3.0).

Figure 1-3. Bits in Port 3 Register

Serial I/O Port

This port allows I/O expansion with UART protocols. The serial port controller (SCON) controls this port. SCON is both bit and byte addressable. Figure 1-4 explains each of the SCON bits. The bit selector to address each of the bits is shown in parenthesis.

BITS

7	6	5	4	3	2	1	0
SM0	SM1	SM2	REN	TB8	RB8	T1	R1

SM0: Serial mode control bit 0 (SCON.7).
SM1: Serial mode control bit 1 (SCON.6).
SM2: Serial mode control bit 2 (SCON.5).
REN: Receiver enable (SCON.4).
TB8: Transmit bit 8 (SCON.3).
RB8: Receive bit 8 (SCON.2).
T1: Transmit interrupt flag (SCON.1).
R1: Receive interrupt flag (SCON.0).

Figure 1-4. Bits in SCON Register

Interrupt Control

The IE (interrupt enable) and IP (interrupt priority) registers control timer and I/O interrupts and priorities. If the IE bit for a device is 1, it can interrupt the processor. The processor will not respond to an interrupt until the instruction being executed has been completed (up to four machine cycles).

When the processor responds, hardware disables interrupts of equal or less priority and vectors to the code location specified for the interrupting device. This location usually contains a jump to a longer service routine. The RETI instruction must be used to return from a service routine, so that interrupts are reenabled. Table 1-4 lists the code locations and predefined labels for interrupting devices. The predefined labels can be used to aid in the placement of I/O routines in code memory.

Table 1-4. Labels and Locations of Interrupting Devices

Predefined Label	Reserved Location	Interrupting Device
RESET	00H	Power on RESET (First instruction executed during power-up).
EXTI0	03H	External interrupt 0.
TIMER0	0BH	Timer 0.
EXTI1	13H	External interrupt 1.
SINT	23H	Serial I/O port.

There are two priority levels for interrupts: 0 or 1. The level 1 has the higher priority and can interrupt a level 0 service routine. A level 0 interrupt has no affect on a level 1 service routine. Interrupts on the same level are disabled. Figure 1-5 explains each of the IE and IP bits. The bit selector to address each of the bits is shown in parenthesis.

INTERRUPT ENABLE BITS

7	6	5	4	3	2	1	0
EA			ES	ET1	EX1	ET0	EX0

EA: Enable all interrupts (IE.7).
Next bit is reserved.
Next bit is reserved.
ES: Enable serial port interrupt (IE.4).
ET1: Enable timer 1 interrupt (IE.3).
EX1: Enable external interrupt 1 (IE.2).
ET0: Enable timer 0 interrupt (IE.1).
EX0: Enable external interrupt 0 (IE.0).

Figure 1-5. Descriptions of Interrupt Enable and Interrupt Priority Bits

INTERRUPT PRIORITY BITS

7	6	5	4	3	2	1	0
			PS	PT1	PX1	PT0	PX0

Bits 7,6, and 5 reserved.
PS: Priority of serial port interrupt (IP.4).
PT1: Priority of timer 1 interrupt (IP.3).
PX1: Priority of external interrupt 1 (IP.2).
PT0: Priority of timer 0 interrupt (IP.1).
PX0: Priority of external interrupt 0 (IP.0).

Figure 1-5. Descriptions of Interrupt Enable and Interrupt Priority Bits (Cont'd)

Reset

When the 8051 is reset, the registers will assume the initial values listed in Table 1-5. The first instruction executed in a program is at ROM location 0. RAM contents is unpredictable at reset.

Table 1-5. Register Values at Reset

Register	Value
Accumulator	00H
Multiplication Register	00H
Data Pointer	0000H
Interrupt Enable	00H
Interrupt Priority	00H
Port 0	0FFH
Port 1	0FFH
Port 2	0FFH
Port 3	0FFH
Program Counter	0000H
Program Status Word	00H
Serial Port Control	00H
Serial I/O Buffer	undefined
Stack Pointer	07H
Timer Control	00H
Timer Mode	00H
Timer 0 Counter	0000H
Timer 1 Counter	0000H

Addressing Modes

Brief descriptions of the addressing modes are included in this chapter. More details on addressing modes and operands is included in Chapter 2.

Indirect Addressing

In indirect addressing, the operand specifies a register containing the memory location to be used in the operation.

Immediate Data

The pound sign (#) immediately in front of an operand signifies it is immediate data. The operand must be a valid assembly-time numeric expression.

Data Addressing

Data addresses are located in on-chip RAM (128 locations) or in one of the hardware register addresses. Data addresses between 0 and 127 are on-chip RAM space; those between 128 and 255 are hardware register locations.

Code Addressing

Three types of instructions require code addressing: relative jumps, absolute 2k page jumps or calls, and long jumps or calls. A relative jump code address must be within −128 to +127 bytes from the first byte of the instruction that FOLLOWS the relative jump. A 2k page jump or call code address is a numeric expression, which the assembler evaluates and encodes in the instruction. The least significant 11 bits of the destination address are placed in the opcode byte and one operand byte. At the time the jump or call is executed, the 11-bit page address replaces the least significant 11 bits in the program counter. This allows a range of 2048 bytes, or any location within the current 2k byte page. A long jump or call code address is a numeric expression, which the assembler evaluates and encodes as a 16-bit value in the instruction. This value replaces the 16 bits in the program counter when the jump or call is executed. With 16 bits, any value (0-64k) is acceptable.

Chapter 2

Rules and Conventions

Introduction

The Model 64000 assembler rules and conventions differ from those of the 8051 assembler to some degree. When using the 64000 assembler, only those rules and conventions discussed here and in Chapter 2 of the Assembler/Linker Reference Manual will be recognized.

Operands

The type and number of operands varies with the mnemonic. The 8051 has six types of operands.

a. Special assembler symbols.

b. Indirect addresses.

c. Immediate data.

d. On-chip data addresses.

e. Bit addresses.

f. Code addresses.

Special Assembler Symbols

Special symbols are reserved by the assembler to designate certain registers as operands. The symbol is encoded in the opcode byte. Table 2-1 lists the symbols with their hardware registers.

Table 2-1. Special Assembler Symbols

Special Symbol	Hardware Register
A	Accumulator.
R0 thru R7	The eight general purpose registers in the current active bank (there are four banks).
DPTR	Data pointer: 16-bit register to index tables in code address and external address spaces.
PC	Program counter.
C	Carry flag receives ALU carry and borrow from bit 7 of the operands.
AB	Accumulator/register B pair for MUL and DIV instructions.

Example:

```
MOV R5,A          ;Move the contents in the
                  ;accumulator to general register
                  ;R5.
```

Indirect Addressing

An indirect address operand specifies a register containing the address of the data to be used in the operation. The address depends on the contents of the register when the instruction is executed. In most cases, the indirect address is in on-chip memory. However, external code or data memory is addressed by the MOVC or MOVX instructions with an indirect address operand. Either R0 or R1 in the active register bank can be specified as the on-chip indirect address operand. A commerce at symbol (@) preceding R0 or R1, or a symbol defined to represent the special register symbols, indicates indirect addressing. The address in the registers must be between 0 and 127. This prevents accessing hardware registers with indirect addressing.

Example:

```
ADD A,@R0         ;Add the contents at the on-chip
                  ;RAM address specified by R0 to
                  ;the accumulator.
```

Immediate Data

The immediate data operand is a numeric expression that the assembler encodes as part of the instruction. A pound sign (#) preceding the expression identifies the operand as immediate data. The numeric expression has to be a valid assembly time expression.

Most instructions require immediate data by one byte. The assembler represents numeric expressions in 16 bits. The low order byte is used and allows the numeric expression to range from a value of −256 to +255. All numeric expression values will have a high order byte with all ones or zeros, when the assembler represents the expression in 16 bits. With those instructions that accept 16-bit values, any value representable by the assembler can be used in the operand. All segment types can be used in the operand (i.e., CSEG, DSEG, XSEG, or BSEG).

Example:

```
MOV A,#0F0H          ;Replace contents of
                     ;accumulator with hex constant F0.
```

Data Addressing

The operand is a numeric expression evaluating to either one of the 128 on-chip memory addresses or one of the hardware register addresses. Only the low-order byte of the 16-bit assembler internal representation is used. Since the 8-bit value in the instruction is unsigned, the value is positive between 0 and 255. In data addressing, the symbol or expression must have the DSEG type or no segment type.

Data addresses between 0 and 127 are in on-chip RAM. Addresses between 128 and 255 are in hardware registers. Some hardware register addresses are reserved; undefined data will be returned from these locations. Refer to the manufacturer's information for details.

Example:

```
MOV P0,A             ;Move the accumulator contents
                     ;to hardware register P0.
```

Bit Addressing

The assembler encodes a bit address in the instruction as a numeric value. A bit address is specified as follows.

a. Define the byte containing the bit with a data address. The bit in the byte is specified with the bit selector (.) followed by the bit identifier (0-7). Refer to the manufacturer's information to determine which on-chip addresses are bit addressable.

   OR

b. Use a numeric expression that evaluates to a bit address. Recall from Data Addressing, the low order byte in the 16-bit assembler internal representation is used. With no sign bit, the value is between 0 and 255.

Instructions with a bit address operand must use symbols or expressions with BSEG or no segment type. Refer to the manufacturer's information for bit addresses that are predefined as symbols.

Examples:

```
SETB 0B0H.4          ;Set bit 4 at location 0B0H
                     ;(timer/counter 0 external
                     ;flag).

SETB T0              ;Set the predefined bit
                     ;address T0 (timer/counter 0
                     ;external flag).
```

Code Addressing

Three types of instructions require code addressing: relative jumps, long jumps or calls, and absolute 2k page jumps or calls. Each differs in the value range of the code address operand. Although all numeric expressions are between 0 and 64k, not all possible addresses are valid for a relative jump or a 2k page jump. Instructions must use symbols or expressions with CSEG or no segment type.

The code address to a relative jump (SJMP and conditional jumps) must be within –128 to +127 bytes from the first byte of the instruction FOLLOWING the relative jump. The assembler computes a relative offset from the specified code address that is encoded as an 8-bit two's complement. The offset is added to the program counter (PC) when the jump is made. Note that because the PC is always incremented to the next instruction before the jump is executed, the range must be computed from the instruction following the relative jump. The numeric expression used must evaluate to the absolute code address of the jump location. All offset computation is done by the assembler. The assembler will report an error if any address is out of range.

The code address to a 2k page jump or call (AJMP and ACALL) is a numeric expression that is evaluated and encoded in the instruction by the assembler. The least significant 11 bits in the page address are contained in the opcode byte and one byte of the operand. During execution of the jump or call, the 11-bit page address replaces the least significant 11 bits in the program counter. This allows addressing anywhere on the current 2k byte page or a range of 2048 bytes. In the case where the jump or call is the last instruction on a 2k page, the five most significant bits in the PC change when the PC is incremented for the next instruction. The jump or call will then be made on a new 2k page.

The code address to a long jump or call (LJMP and LCALL) is a numeric expression that the assembler evaluates and encodes as a 16-bit value in the instruction. The contents of the program counter are replaced by this value when the jump or call is executed. With 16 bits, any value represented by the assembler is acceptable (0-64k).

Examples:

```
SJMP TABLE          ;Jump to TABLE (TABLE is between
                    ;–128 and +127 of instruction
                    ;following SJMP).

ACALL FILL          ;Call subroutine FILL (FILL is an
                    ;address within the current 2k
                    ;page).

LJMP NEXT           ;Long jump (NEXT must be defined in
                    ;the program).
```

The two instructions: JMP or CALL do not assemble to a specific opcode. JMP can assemble to jump instruction SJMP, AJMP, or LJMP. CALL can assemble to ACALL or LCALL. JMP or CALL always evaluates to an instruction that reaches the specified code address operand.

Symbols

To increase readability, the programmer can define symbols for code, bit, or data addresses; constants; or registers. The symbols are assigned two attributes when they are defined in the program: numeric value and segment type. The segment types are as follows:

BSEG-Bit address space
CSEG-Code address space
DSEG-Data address space
XSEG-External data address space

After a symbol is defined, it can be used in any numeric operand as a constant, respecting segment type conventions.

In addition to these user-defined symbols, symbols are predefined for commonly used hardware registers and flags. These symbols were explained in Chapter 1, table 1-2; they equate to a data address and cannot be used with instructions requiring a special symbol for the hardware register or flag.

Examples:

```
ADD A,#9          ;Valid instruction, A is correct
                  ;special symbol for this operand
                  ;and instruction.

ADD ACC,#9        ;Incorrect, ACC is predefined
                  ;symbol for an address.
```

Special Operators

The special operators HIGH and LOW select the high order byte or low order byte in the operand.

Examples:

```
HIGH 0A72BH       ;Selects the high order byte of
                  ;0A72BH.

LOW 0A72BH        ;Selects the low order byte of
                  ;0A72BH.
```

Predefined Symbols

The following symbols are reserved. They have special meaning to the assembler and cannot appear as user defined symbols.

Reserved Words

A	AB	C	DPTR	PC	R0
R1	R2	R3	R4	R5	R6
R7					

Predefined Data (Byte) Addresses

ACC	B	DPH	DPL	IE	IP
P0	P1	P2	P3	PSW	SBUF
SCON	SP	TCON	TH0	TH1	TL0
TL1	TMOD				

Predefined Bit Addresses

AC	CY	EA	ES	ET0	ET1
EX0	EX1	F0	IE0	IE1	INT0
INT1	IT0	IT1	OV	P	PS
PT0	PT1	PX0	PX1	RB8	RD
REN	RI	RS0	RS1	RXD	SM0
SM1	SM2	T0	T1	TB8	TF0
TF1	TI	TR0	TR1	TXD	WR

Predefined Code Addresses

RESET	TIMER0	TIMER1	EXTI0	EXTI1	SINT

Chapter 3

Special Pseudo Instructions

Introduction

This chapter supplements Chapter 3 in the HP Model 64000 Assembler/Linker Reference Manual. It lists and defines in detail those assembler instructions that are applicable to the 8051 microcomputer only.

Assign Bit Address

SYNTAX:

Label	Operation	Operand	Comment
symbol	BIT	address	

The BIT instruction assigns a bit address to symbol. No forward references are allowed. A symbol defined with BIT will have a BSEG segment type and cannot be redefined elsewhere in the program.

Example:

Label	Operation	Operand	Comment
TIME_ERR	BIT	TF1	

This statement assigns the predefined bit address TF1 to TIME_ERR.

BSEG

Selects Bit Address Segment

SYNTAX:

Label	Operation	Operand	Comment
[name]	BSEG		

This statement selects the bit address segment in the 8051. Each BSEG restores the location counter to its value when the bit segment was last active. The location counter can be altered with the ORG and DBIT pseudo instructions. Every unit of the location counter represents a bit in the bit address space. The values in the location counter range from 0 to 255.

Example:

Label	Operation	Operand	Comment
SAM	BSEG		

Selects Code Address Segment

SYNTAX:

Label	Operation	Operand	Comment
[name]	CSEG		

The CSEG instruction selects the code address segment in the 8051. This is the default segment when the assembler is invoked. Within the code segment there are three counters: program relocatable, common, and absolute. CSEG invokes the program relocatable counter and restores it to the value it had when it was last active. COMN invokes the common counter and restores it to the value it had when it was last active. ORG invokes the absolute counter and starts absolute code.

The code segment counters can be changed with the DS, DB, and DW pseudos and with each instruction encoded. Each unit in the program relocatable counter represents one byte in the code address space within the range of 0 to 64k.

Example:

Label	Operation	Operand	Comment
SAM	CSEG		

Assigns an On-chip Data Address

SYNTAX:

Label	Operation	Operand	Comment
symbol	DATA	expression	

This instruction assigns an on-chip data address to symbol. The expression must be a valid assembly time expression with no forward references that evaluates to a data address (0-255). The symbol will be segment type DSEG and cannot be redefined in the program.

Example:

Label	Operation	Operand	Comment
LIST	DATA	65H	

This statement assigns the on-chip address 65H to LIST.

Selects Code Address Segment

SYNTAX:

Label	Operation	Operand	Comment
[name]	DSEG		

This statement selects the on-chip data address segment. Each DSEG restores the location counter to its value when the data segment was last active. The location counter can be altered with the ORG and DS pseudo instructions. Every unit of the location counter represents one byte in the on-chip data address space. The values in the location counter range from 0 to 255.

Example:

Label	Operation	Operand	Comment
SAM	DSEG		

Reserves Bit Address Space

SYNTAX:

Label	Operation	Operand	Comment
[name]	DBIT	expression	

This statement reserves bit address space specified by expression. It is only used in the bit address segment. Expression must be a valid assembly time expression with no forward references. DBIT causes the location counter of the bit address segment to be incremented by the value of expression.

Example:

Label	Operation	Operand	Comment
name	DBIT	SAM	

Initializes Code Memory with Byte Values

SYNTAX:

Label	Operation	Operand	Comment
[name]	DB	expression list	

This statement initializes code memory with byte values: –256 to +255. Consequently, the code segment must be active. Expression list is one or more byte values or strings separated by commas. A byte value is an assembly time expression or a character string. Each byte value is stored in memory in the same order it appears in the expression list. If a label is used, its value is the address of the first byte in the list.

Example:

Label	Operation	Operand	Comment
	DB	"SAM"	

Initializes Code Memory with Word Values

SYNTAX:

Label	Operation	Operand	Comment
[name]	DW	expression list	

This statement initializes code memory with a list of word values (16 bits). The code segment must be active. Expression list is one or more word values separated by commas. A word value is an assembly time expression or a character string. If the location counter ($) is used in the expression list, it is evaluated as a code address in the high order byte of the word being initialized. Each word value is stored in memory in the same order it appears in the expression list. If a label is used, the label value addresses the high order byte of the first word in the list.

Example:

Label	Operation	Operand	Comment
WTS	DW	31,46,77,89	

Reserves Space in Bytes

SYNTAX:

Label	Operation	Operand	Comment
[name]	DS	expression	

This statement reserves space in bytes and can be used in any segment except bit address. Expression must be a valid assembly time expression with no forward references. DS causes the location counter of the current segment to be incremented by the value of expression. The sum of the location counter and expression cannot exceed the limitations of the current address space.

Example:

Label	Operation	Operand	Comment
	DS	SAM	

Assign Value or Symbol to Symbol Name

SYNTAX:

Label		Operation	Operand	Comment
symbol		SET	expression	
	or			
symbol		SET	symbol	

This statement assigns a numeric value or special assembler symbol (see Chapter 2) to a symbol name. Expression must be a valid assembly time expression with no forward references. The special assembler symbols A and R0 through R7 can be represented by user symbols defined with SET. A symbol defined with SET has no segment type. It can be used as a bit address, code address, data address, or external data address. A symbol defined to a register value will have the type "REG". After this point, the symbol can only be used instead of the register in operands. A symbol can be redefined later with the SET instruction.

Example:

Label	Operation	Operand	Comment
SAM	SET	R3	

Assign Off-chip Data Address to Symbol

SYNTAX:

Label	Operation	Operand	Comment
symbol	XDATA	expression	

This statement assigns an off-chip data address to symbol. Expression must be a valid assembly time expression with no forward references. The symbol will have an XSEG segment type. The symbol cannot be redefined later in the program.

Example:

Label	Operation	Operand	Comment
SAM	XDATA	999H	

Select External Data Address Segment

SYNTAX:

Label	Operation	Operand	Comment
[name]	XSEG		

This statement selects the external data address segment. Each XSEG restores the location counter to its value when the external data segment was last active. The location counter can be altered with the ORG and DS pseudo instructions. Every unit of the location counter represents one byte in off-chip data address space. The values in the location counter range from 0 to 64k.

Example:

Label	Operation	Operand	Comment
SAM	XSEG		

MP MASTER
MA27283

64855-90903 DECEMBER 1980

HEWLETT PACKARD

PRINTED IN U.S.A.